2001

VEMECS: A virtual reality interface for spherical mechanism design

Juliet Kraal
Ford Motor Company

Judy M. Vance
Iowa State University, jmvance@iastate.edu

VEMECS: A virtual reality interface for spherical mechanism design

Abstract

Mechanisms are one of the fundamental elements used by engineers in the design of machines. The design or synthesis of these elements has been studied for hundreds of years. More recently, the focus of kinematic synthesis research has been in reformatting the graphical synthesis methods into an analytical form compatible with computer processing. One of the thrusts of this research concentrates on the human/computer interface (HCI) between the user and the computer design software. The work presented in this paper addresses this issue of developing a new HCI for mechanism design based on virtual reality techniques. Current computer-aided mechanism design can be seen in the areas of analysis, topological, and dimensional synthesis (Erdman 1995). In each of these areas, the trend is toward developing user interfaces that are more compatible with the cognitive and perceptual nature of the designer. Mechanism synthesis computer programs like KINSYN (Rubel et al . 1977), LINCAGES-4 (Erdman and Riley 1981) and SPHINX (Larochelle et al . 1993) utilize the traditional HCI of monitor, keyboard and mouse. More recently, virtual reality (VR) is being examined as a natural human interface for mechanism design. Osborn (1994) developed the first virtual environment for the synthesis of spherical four-bar mechanisms, SphereVR, which used the Newton-Raphson iterative approach to solve the design equations. This paper takes a slightly different approach by combining the reliable and robust solution algorithms of SPHINX1.0 with VR technology. VR is used for all interaction, manipulation, and navigation throughout the design process, while SPHINX1.0 computational routines are used for computing the solution mechanisms. Natural and intuitive skills of the designer are used through reliance on a head tracked three-dimensional display and three-dimensional interaction. The result is a program called VEMECS (Virtual Environment MEChanism Synthesis). This paper outlines the organization and operation of VEMECS, and concludes with a discussion of the lessons learned in development and implementation of this approach.

Keywords

Iowa Center for Emerging Manufacturing Technology

Disciplines

Computer-Aided Engineering and Design | Graphics and Human Computer Interfaces | Manufacturing

Comments

This is an Accepted Manuscript of an article published by Taylor & Francis in Journal of Engineering Design in 2001, available online: http://www.tandf.com/10.1080/09544820110055574.

This is an Accepted Manuscript of an article published by Taylor & Francis in Journal of Engineering Design in 2001, available online: http://www.tandf.com/10.1080/09544820110055574.

VEMECS: A Virtual Reality Interface for Spherical Mechanism Design

Juliet C. Kraal
Ford Motor Company

Judy M. Vance
Iowa Center for Emerging Manufacturing Technology
2026 Black Engineering Building
Iowa State University
Ames, IA 50011
jmvance@iastate.edu
ph: 515-294-9474
fax: 515-294-3261

Mechanisms are one of the fundamental elements used by engineers in the design of machines. The design or synthesis of these elements has been studied for hundreds of years. More recently, the focus of kinematic synthesis research has been in reformatting the graphical synthesis methods into an analytical form that is compatible with computer processing. One of the thrusts of this research concentrates on the human/computer interface (HCI) between the user and the computer design software. The work presented here addresses this issue of developing a new HCI for mechanism design based on virtual reality techniques. Current computer-aided mechanism design can be seen in the areas of analysis, topological, and dimensional synthesis (Erdman, 1995). In each of these areas the trend is toward developing user interfaces that are more compatible with the cognitive and perceptual nature of the designer. Mechanism synthesis computer programs like KINSYN (Rubel and Kaufman, 1977), LINCAGES-4 (Erdman and Riley, 1981) and SPHINX (Larochelle, et.al., 1993)utilize the traditional HCI of monitor, keyboard and mouse. More recently, VR is being examined as a natural human interface for mechanism design. Osborn (1994) developed the first virtual environment for the synthesis of spherical four-bar mechanisms, SphereVR, which used the Newton-Raphson iterative approach to solve the design equations. This paper takes a slightly different approach by combining the reliable and robust solution algorithms of SPHINX1.0 with VR technology. VR is used for all interaction, manipulation, and navigation throughout the design process, while SPHINX1.0 computational routines are used for computing the solution mechanisms. Natural and intuitive skills of the designer are used through reliance on a head tracked three-dimensional display and three-dimensional interaction. The result is a program called VEMECS (Virtual Environment MEChanism Synthesis). This paper outlines the organization and operation of VEMECS and concludes with a discussion of the lessons learned in development and implementation of this approach.

1 INTRODUCTION

In the early days of computing, punch cards were used to input information into the computer and printed text was used for computer output. Over the years, this type of human/computer interaction has been replaced with computer monitors and keyboards. In the early 80s, Apple Computer introduced a graphical user interface consisting of icons, or small pictures which introduced the desktop mouse as an input (Rheingold, 1991). The implementation of this graphical user interface as a means of communicating with the computer takes advantage of the human capability to recognize and process graphical images quickly and has become a universal HCI standard.

Virtual reality (VR) technology attempts to extend the traditional HCI even further in an attempt to provide an interface that removes the barriers of communication between the human and the information stored in the computer. To accomplish this, VR capitalizes on the unique capabilities of the human senses and on how the brain integrates information gathered from these senses. VR applications utilize various computer peripherals to interface with the human senses in a manner entirely different from the monitor, keyboard and mouse. This technology has given birth to new and innovative computer interaction devices. As these devices have matured, industries have started to investigate using VR technology for engineering design applications (Mahoney, 1995)(Schmitz, 1995). The goal of the work presented here is to investigate the application of VR to spherical mechanism design.

Burmester (1888) developed a library of geometric constructions for the synthesis of planar mechanisms. This work resulted in a graphical method for designing mechanisms and is often recounted as the first systematic treatment of mechanisms. In 1959, Freudenstein and Sandor (1959) presented a method that transformed Brumester's graphical-based linkage synthesis techniques into analytical equations that could be solved by a computer. Originally computers seemed best suited to the kinematic analysis of mechanisms. Batch processes were used to calculate the characteristic displacements, velocities, and accelerations of the mechanism. In the 1970's interactive computing was introduced and relieved the designer of the burden of batch computing. During the early 1970's computer graphics were first applied to mechanism design by Kaufman in the software KINSYN I which was developed at M.I.T. (Kaufman, 1978). The 1980's were filled with the design and marketing of several other mechanism design packages. During this time the kinematic synthesis and analysis tools were integrated and mechanism design was combined with other design tools such as computer-aided drafting, finite-element analysis (FEA), and simulation (Erdman and Sandor, 1991).

SPHINX1.0 was introduced in 1993 by McCarthy and Larochelle (Larochelle, et.al., 1993) as a tool for the design of spherical four-bar mechanisms. SPHINX1.0 is workstation-based and presents the user with a series of pull down menus much like other common human computer interfaces present in mechanical design software. It is the computation routines of SPHINX1.0 that form the heart of the VEMECS software. In 1997, SPHINXPC was introduced for the personal computer platform (Ruth and McCarthy, 1997).

2 PROGRAM DEVELOPMENT

VEMECS was written in C and developed using the WorldToolKit (WTK) virtual reality software and Silicon Graphics Graphic Libraries (GL). WTK is a set of computer graphics functions developed by the Sense8 Company that allow a wide array of VR peripherals to be integrated into a virtual reality application. The SPHINX1.0 software was also written in C and GL. Three main issues were encountered during the development of the VEMECS software: integration, interaction, and immersion. These issues are discussed in the following sections.

2.1 Integration

The integration of the SPHINX code into a WTK program required a correlation of the world coordinate systems used by each software package. Both WTK and SPHINX use a right-handed cartesian world coordinate system. SPHINX, however, defines +Y pointing upward and WTK's default universe coordinate system defines +Y pointing downward (Figure 1). The difference between the two systems is a 180 degree rotation about the X axis. This difference in the world coordinate systems is most obvious when users interact within the VE. The discrepancy is a result of the visual universe being drawn in SPHINX's coordinate system and the interaction with the universe calculated in WTK's coordinate system. This inconsistency was addressed by a rotation of the GL code by 180 degrees about the X axis.

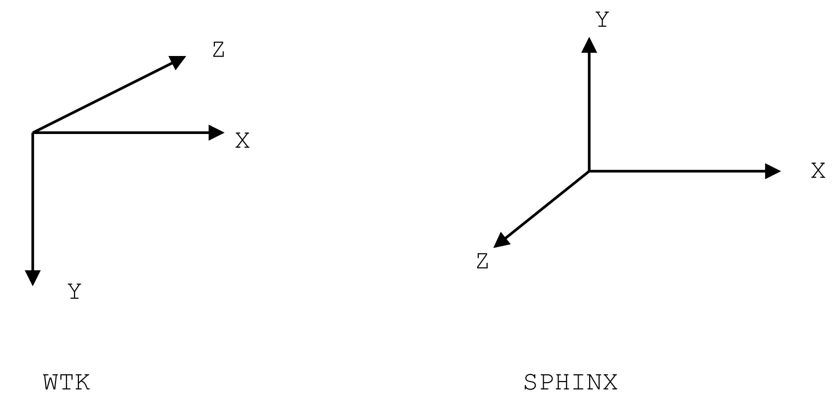

Figure 1: Difference between WTK and SPHINX1.0 world coordinate systems

Another interface issue dealing with the differences in coordinate systems emerged during the task of placing of prescribed positions on the design sphere. The HCI of SPHINX is confined to the two dimensional workstation display, keyboard, and mouse. The specification of the positions on the sphere, therefore, are performed using two-dimensional input. The combination of the 2D mouse movement on the screen and a specific button press alters the longitude, latitude, and roll of default positions shown on the design sphere. VEMECS, on the other hand uses a three-dimensional interface device, and does not have a set of default positions specified on the design sphere.

In order to provide three-dimensional cues to the user, a cursor-like object is used in the VE to specify the orientation and placement of desired positions on the design sphere. This cursor-like object uses a right-handed cartesian coordinate system, and displays +X, +Y, and +Z (Figure 2). When the position is specified, the +X, +Y, and -Z axes are drawn. The reason for displaying -Z is that it represents an axis that passes through the center of the sphere. The orientation of this cursor is obtained through a WTK function that returns its orientation in quaternion representation. This orientation is with respect to WTK's coordinate system and it must be changed into the mechanism synthesis orientation before calculations are performed. Details of this transformation can be found in Kraal, 1996.

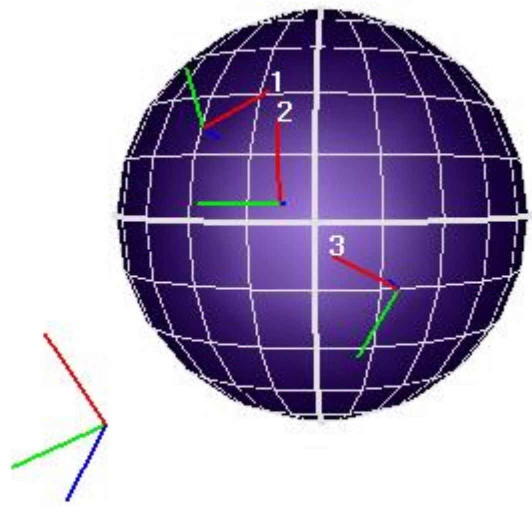

Figure 2: VEMECS position cursor

2.2 Interaction

Interaction with SPHINX1.0 is performed using a desktop mouse. Two types of interaction devices have been incorporated into VEMECS. The first method uses a Logitech 6D Mouse as the communication device between the designer and the environment. The Logitech is a position tracker as well as a communication device. To indicate to the designer the location of this device, an virtual arrow or "pointer" follows the device's motion in the VE. This pointer is used for selecting objects, moving objects, and making menu choices. As the pointer intersects with menu items the menu item changes color. Buttons on the 6D mouse are used to interact with the computer images. This three-dimensional mouse-type interaction can be quite natural for persons with computer experience who are accustomed to using the mouse buttons as input to the computer.

The second interaction device that can be used in VEMECS is the CyberGlove^{TM}. The CyberGlove^{TM} is an instrumented glove that senses the position of each finger and the thumb and then manipulates a computer model of the hand shown in the VE. This can be used as an effective extension of the designer's hand into the VE. The communication with the environment is done through natural hand gestures. For example, to grasp an object such as the design sphere, the designer simply presses the index finger and thumb together while they intersect the design sphere.

Then, as the hand is moved, the object is also moved. To release the object, the user releases the virtual object by separating the index finger from the thumb. This extension of the designer's hand and the use of natural gestures provide an intuitive first person method of communication between the designer and the design space.

The menus used in the mechanism synthesis of VEMECS are an important part of the interaction. Although menus are not necessary while interacting with the physical world, some abstract tasks are more effectively performed through the assistance of menus when interacting within a virtual environment. The menu system used in VEMECS consists of WTK text objects. It is designed for a user, knowledgeable in mechanism synthesis, who occasionally designs spherical mechanisms. The menu text follows the designer's movement and can be toggled into and out of sight (Figure 3). It can be accessed with either of the interaction devices: the Logitech 6Dmouse or the CyberGlove™ hand. These menus and their interactions provide an appropriate interactive environment for mechanism synthesis.

Figure 3: VEMECS main menu

2.3 Immersion

Immersion is an important aspect of VR and is defined as making the user feel as if he/she were a part of the computer generated environment (Aukstakalnis and Blatner, 1992). The Virtual Research Eyegen3 HMD, Fakespace BOOM3C, stereo monitor, and single stereo projection wall are all devices that have been integrated to achieve visual immersion within VEMECS.

The HMD or "helmet" is a visually immersive VR display, which when worn on the head, effectively blocks out the view of the real world and only presents the computer world to the user. The Fakespace BOOM3C (Binocular Omni-Oriented Monitor) consists of a CRT monitor supported by a mechanical arm and tracked with optical encoders at the arm joints. Since the CRT monitor is not worn on the head, higher resolution can be obtained in the BOOM without regard to weight of the device. To view the application using the single stereo projection wall, the users wear CrystalEyes stereo glasses. These are active shutter glasses that provide a three-dimensional image to the user. This approach is not totally immersive because the user is still able to see the real world through the glasses.

3 PROGRAM OPERATION

The VEMECS program performs four-position motion generation for spherical mechanism design. The steps in VEMECS that are needed to create a spherical mechanism involve sizing the constraint sphere, specifying the four design positions and related orientations, specification of the driving and driven links, generating the mechanism and verifying the mechanism motion. Each of these steps will be explained in the following section.

3.1 Sizing the sphere

VEMECS presents four menu categories in the main menu: *Synthesis*, *Settings*, *Simulation* and *File*. The *Synthesis* menu option is where most of the mechanism design tasks are performed. The first step is to size the design sphere. All of the desired design positions will be placed on this sphere. The size of the design sphere will be determined relative to the scale of the project. Since all of the motion of the mechanism is constrained to lie on concentric spheres about the center of the design sphere, the size of the linkage is scalable. Choosing the *Size* causes the display of the sphere to change to a wireframe model with size markers highlighted. These markers can be moved to change the size of the sphere (see Figure 4). Once these markers are "released" the sphere is returned to the solid representation and the size markers are removed. After the sphere has been appropriately sized, the positions can be specified.

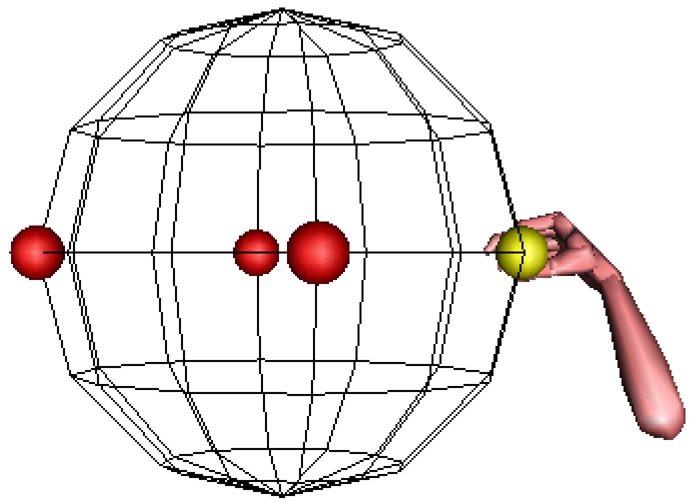

Figure 4: Sizing the design sphere

3.2 Specification of positions and orientations

To specify the four prescribed positions for the coupler link, the designer chooses the menu option numbers 1, 2, 3, and 4 for each respective position. Position specification is unique to the type of interaction device used in VEMECS. As the Logitech pointer activates each number, the pointer is transformed into a triad. This triad is now tracked by the Logitech and can be placed on the sphere. As the triad intersects with the sphere the triad is aligned with the Z axis along an axis from the center of the sphere to the point of intersection. When the triad comes into contact with the sphere a yellow ball appears to indicate the contact as shown in Figure 5. Using the Logitech tracker the Z axis can be positioned on the sphere's surface and the remaining X and Y axes can be rotated about this axis by rotating the Logitech pointer. The CyberGlove™ interaction differs slightly from the Logitech pointer interaction in that as the menu number is activated, the triad is "attached" to the index finger of the virtual hand. This can be placed and orientated with the same interaction as described with the Logitech pointer.

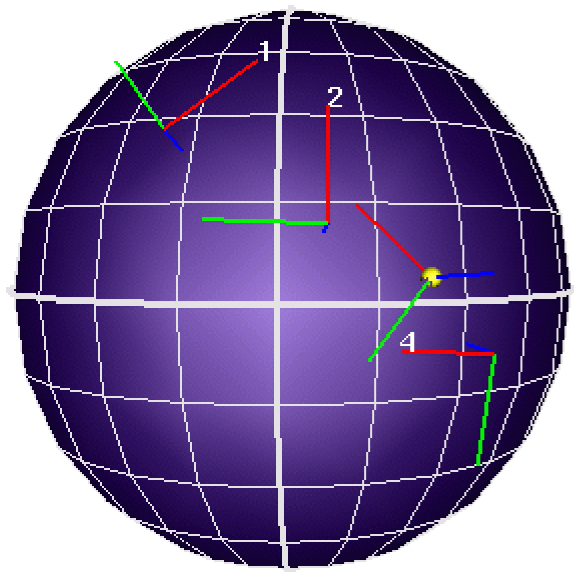

Figure 5: Placing positions on the sphere

3.3 Specification of the driving and driven links

Once all four positions have been specified the circle-axis and center-axis cones can be generated. This is done by choosing *Axes* in the *Synthesis* menu. As this option is chosen, the cones are generated and displayed on the design sphere. The circle-axis cone is shown in yellow and represents all of the solution locations of the moving axes. The fixed-axis cone is shown in red and represents all of the solution locations of the fixed axes. Next, the *Axes* menu appears. Interaction with this menu allows the specification of the driving and driven links by selecting axes on the circle-axis and center-axis cones. There is a *Preference* menu that provides the user the option to designate which axes to display: fixed (center-axis cone), moving (circle-axis cone), or both. The driving and driven links are specified by intersecting the Logitech pointer or the virtual index finger with an axis on the cone. The choice of a fixed axis results in choosing the corresponding moving axis, thus defining a link. The chosen driving link axes are displayed in green while the driven link axes are shown in red (see Figure 6). After a link has been specified, both the moving and fixed axes can be negated. This results in selecting the opposite intersection of the axis with the design sphere as the chosen axis. This can be done with the toggles *nfixed*, which negates the fixed axis of the link being specified, and *nmoving* which negates the moving axis of the specified link.

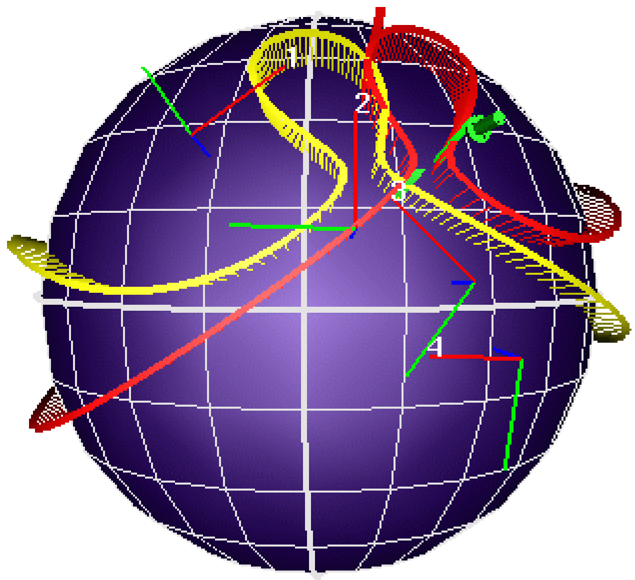

Figure 6: Selecting the driving and driven links

3.4 Generating the mechanism

Once the driving and driven links have been selected, the mechanism can be generated. To generate the mechanism, the *Mechanism* option is chosen from the *Synthesis* menu. This displays the mechanism that has been designed and its associated coupler curve. The coupler curve is the path that the point of interest on the coupler link will traverse as the mechanism moves. The coupler link's orientation as it moves along this path is illustrated with a triad that indicates the rotation about an axis through the center of the sphere. An example of a mechanism design in VEMECS is shown in Figure 7. The mechanism is shown mounted on a representative "base".

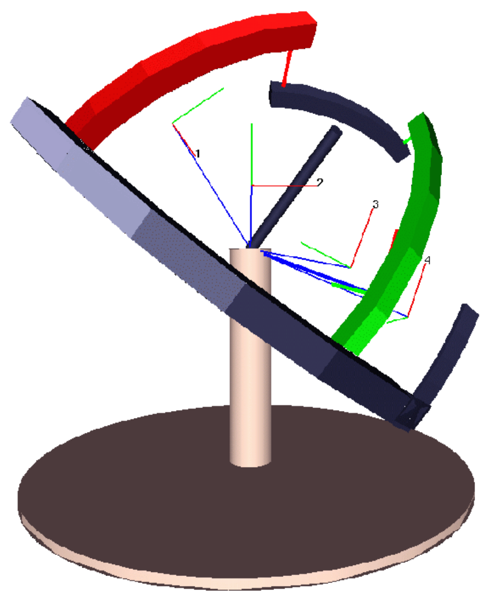

Figure 7: The final mechanism and base

3.5 Mechanism verification

Verification of the design is possible by examining the motion of the mechanism and the shape of the coupler curve to ensure there are no order or branch defects. The *Simulation* menu allows the user to animate the mechanism. *Move/Stop*, *Reverse,* and *Faster* are all options that control the animation of the mechanism. A mechanism that travels through positions 1, 3, 2, and then 4 is an example of an order defect. Branch defects occur when the rigid members of the mechanism physically prevent the mechanism from following the desired path, thus resulting in a folded mechanism. Both of these defects result in a mechanism that will not satisfy the specified motion even though the coupler link passes through the desired positions. Further mechanism validation can be accomplished by viewing the mechanism within its design context.

3.6 Setting options

Several different preference settings are provided under the menu option *Settings*. All of these settings are toggles that either turn an option on or off or switch it between two options. The *Attachment*, *Curve*, and *Axes* menu items toggle the coupler attachment, coupler curve, and design sphere axes on and off. The coupler attachment is the piece that is attached to the coupler that locates the coupler point with respect to the coupler link. The *Solid/Wireframe* toggles the display between a solid representation and a wireframe representation. The *Sphere/Null/Base* toggles each option between showing the conceptual design sphere, showing only the mechanism, and showing the base. A default base has been designed that can replace the design sphere. The *Null* option is most useful when viewing the mechanism within its design context. These options give the designer a more intuitive view of the mechanism in different application configurations.

3.7 File options

There are several file options that allow efficient storage and recall of spherical mechanisms designed in VEMECS. Many of these options parallel the file menus in traditional windowing interfaces. *New*, *Open*, *Save, Saveas,* and *Exit* options are analogous to the familiar uses of these menu options in traditional two dimensional windowing applications. The *save* option writes out two different text files that can be used to save the mechanism specification.

4 RESULTS

Several observations were made based on experience with this VR environment:

1. The addition of stereo display greatly enhanced the ability to visualize the final motion of the solution. Stereo display is not a part of the traditional workstation-based design software but is one characteristic of VR.
2. The three-dimensional interaction devices provided a very intuitive and easy method for specifying spatial information. Placing desired positions of the coupler point and orientations of the coupler link on the sphere is a very difficult task in traditional workstation-based spherical mechanism design software.
3. A menu system was used in this virtual environment. It was more efficient to be able to select position 1 from a menu than if the user would have had to pick up an object representing position 1 from someplace in the virtual world.
4. Heads-up display of the menus was effective. Even when the viewpoint was changed, the menu remained in the viewing area. A result of this was the need to hide the menu at times and this was accomplished using gestures or mouse buttons.
5. Head-coupled viewing provided by the HMD or the BOOM was effective in providing the designer with a spatial sense of the design space. Viewing the virtual environment on a monitor or on a projection screen without head coupled viewing did not provide this spatial awareness.

5 CONCLUSIONS

A virtual environment called VEMECS was developed to provide a virtual design space for the synthesis of spherical mechanisms. The analytical basis of spherical mechanism design was successfully incorporated into a virtual design environment. Various visual displays, including an HMD, stereo glasses, and the BOOM were integrated into the VR application. Interaction devices within the VE include the Logitech 6D mouse and the CyberGlove™. These interaction devices are used to access menus and input design parameters in three-dimensional space.

VEMECS provides an effective design space for the synthesis of spherical mechanisms. This environment combines the reliable and robust solution algorithms of SPHINX1.0 with VR technology. VR is used for all interaction, manipulation, and navigation throughout the design process. Natural and intuitive skills of the designer are used through reliance on a head-coupled display and three-dimensional interaction. The VEMECS environment illustrates a successful application of VR technology to conceptual design.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the cooperation of Professor J. Michael McCarthy at the University of California, Irvine for allowing access to the SPHINX1.0 computational routines and providing technical support with regard to those routines. This work was funded in part by Iowa State University and the Iowa Center for Emerging Manufacturing Technology.

REFERENCES

Erdman, A. G., Computer-Aided Mechanism Design: Now and the Future, ASME Journal of Mechanical Design, vol. 117(B), June 1995, 93-100.

Erdman, A. G., Riley, D., Computer-aided Linkage Design Using the LINCAGES Package, Proceedings of the 1981 ASME Design Technical Conferences, 81-DET-121, 1981.

Larochelle, P., Dooley, J., Murray, A., McCarthy, J.M., SPHINX-Software for Synthesizing Spherical Mechanisms, Proceedings of the 1993 NSF Design and Manufacturing Systems Conference, Charlotte, NC, January 6-8, vol. 1, 607-612.

Rubel, A. J., Kaufman, R.E., Kinsyn III: A New Human-engineered System for Interactive Computer-aided Design of Planar Linkages, ASME Journal of Engineering for Industry, vol. 99, May 1977, 440-448.

Osborn, S. W., Applying virtual reality to aid in the synthesis of spherical four-bar mechanisms, M.S. Thesis, Iowa State University, Ames, IA, 1994.

Rheingold, H., Virtual Reality, New York, NY: Simon & Schuster, 1991.

Mahoney, D. P., Driving VR, Computer Graphics World, vol. 18, no. 5, 1995, 22-33.

Schmitz, G., Great Expectations: The Future of Virtual Design, Computer-Aided Engineering, vol. 14, no. 10, pp. 68-72.

Burmester, L., Lehrbuch der Kinematik. Leipzig, Germany, 1888.

Freudenstein, F., Sandor, G. N., Synthesis of Path-generating Mechanisms by Means of a Programmed Digital Computer. ASME Journal of Engineering for Industry, vol 81B, May 1959, 159-168.

Kaufman, R. E., Mechanism Design by Computer, Machine Design, October 1978, 94-100.

Erdman, A. G., Sandor, G. N., Mechanism Design Analysis and Synthesis vol. 1, Englewood Cliffs, NJ: Prentice Hall, 1991.

Ruth, D. A. and McCarthy, J. M., SphinxPC: An Implementation of Four Position Synthesis for Planar and Spherical 4R Linkages, Proceedings of the 1997 ASME Design Engineering Technical Conferences, September 1997.

Kraal, J. C., An Application of Virtual Reality to Engineering Design: Synthesis of Spherical Mechanisms, M. S. Thesis, Iowa State University, Ames, IA, 1996.

Aukstakalnis, S., Blatner, D., Silicon Mirage: The Art and Science of Virtual Reality, Peachpit Press, 1992, 26.